Europäisches Patentamt | European Patent Office | Office européen des brevets

Bescheinigung | Certificate | Attestation

Die angehefteten Unterlagen stimmen mit der ursprünglich eingereichten Fassung der auf dem nächsten Blatt bezeichneten europäischen Patentanmeldung überein.

The attached documents are exact copies of the European patent application described on the following page, as originally filed.

Les documents fixés à cette attestation sont conformes à la version initialement déposée de la demande de brevet européen spécifiée à la page suivante.

Patentanmeldung Nr. | Patent application No. | Demande de brevet n°

00480076.9

Der Präsident des Europäischen Patentamts;
Im Auftrag

For the President of the European Patent Office

Le Président de l'Office européen des brevets
p.o.

I.L.C. HATTEN-HECKMAN

DEN HAAG, DEN
THE HAGUE, 14/09/00
LA HAYE, LE

Europäisches Patentamt
European Patent Office
Office européen des brevets

Blatt 2 der Bescheinigung
Sheet 2 of the certificate
Page 2 de l'attestation

Anmeldung Nr.:
Application no.:
Demande n°: 00480076.9

Anmeldetag:
Date of filing: 09/08/00
Date de dépôt:

Anmelder:
Applicant(s):
Demandeur(s):
INTERNATIONAL BUSINESS MACHINES CORPORATION
Armonk, NY 10504
UNITED STATES OF AMERICA

Bezeichnung der Erfindung:
Title of the invention:
Titre de l'invention:
System for transmitting LAN data frames through an ATM crossbar switch

In Anspruch genommene Priorität(en) / Priority(ies) claimed / Priorité(s) revendiquée(s)

Staat: State: Pays:	Tag: Date: Date:	Aktenzeichen: File no. Numéro de dépôt:

Internationale Patentklassifikation:
International Patent classification:
Classification internationale des brevets:
/

Am Anmeldetag benannte Vertragstaaten:
Contracting states designated at date of filing: AT/BE/CH/CY/DE/DK/ES/FI/FR/GB/GR/IE/IT/LI/LU/MC/NL/PT/SE
Etats contractants désignés lors du depôt:

Bemerkungen:
Remarks:
Remarques:

SYSTEM FOR TRANSMITTING LAN DATA FRAMES THROUGH AN ATM CROSSBAR SWITCH

Technical field

The invention relates to the transmission of data frames between Local Area Networks (LAN) interconnected by a switch engine and relates in particular to a system for transmitting LAN data frames through an ATM crossbar switch.

Background

Local Area Networks (LAN) such as Ethernet or token-ring networks, are generally interconnected through hubs. The hub is a system made of LAN adapters that communicate together through a switch card. This switch card can be either a parallel bus or a passive switch card including a matrix for achieving the connection between selected inputs and outputs.

Today, the Asynchronous Transfer Mode (ATM) technology is growing very fast and most of the research developments are

concentrated in high speed ATM networks instead of LANs. Very high speed ATM switches are now available and used for transferring data from two LANs connected to the switch. The use of the ATM technology for switching LAN frames requires to transform each LAN frame in splitting the frame into ATM frames by encapsulating them in the ATM Adaptation Layer (AAL) format. For this, the LAN frame coming from each line into the LAN adapter, with its destination address, is transformed onto AAL by a special module into ATM data packets before being transferred to the switch card for switching. Such a requirement results in two important drawbacks. Since the frame is converted into ATM cells, it is necessary to have a header in each cell containing protocol information such as the destination address. The same information is therefore repeated in each ATM cell, which represents many bytes that do not contain data and therefore there is an important waste of bandwidth. A further drawback is that the transformation of the LAN frame into ATM cells and the encapsulation in the AAL format require important hardware and software thereby resulting in a very important cost.

Summary of the invention

Accordingly, the object of the invention is to provide a system for exchanging data between a plurality of Local Area Networks (LAN) interconnected through an ATM crossbar switch without transforming the LAN data frames into ATM cells using the ATM adaptation Layer (AAL) technology.

The invention relates therefore to a data transmission system comprising a plurality of Local Area Networks LANs interconnected by a hub including a plurality of LAN adapters respectively connected to the LANs and an ATM crossbar switch interconnecting all LAN adapters wherein at least one LAN wants to transmit a LAN data frame to another LAN, this LAN data frame being converted into concatenated slots of the same

size and being transmitted through the ATM crossbar switch. At least the LAN adapter connected to the LAN wanting to transmit LAN data frame comprises a serial communication controller including means for converting the LAN data frame into serial data having the form of concatenated slots of the ATM cell size in the HDLC format before transmitting this serial data to the ATM crossbar switch and means for converting serial data having the form of concatenated ATM cells received from the ATM crossbar switch into a LAN data frame before transmitting this LAN data frames to the LAN.

Brief description of the drawings

The above and other objects, features and advantages of the invention will be better understood by reading the following more particular description of the invention in conjunction with the accompanying drawings wherein :

Fig. 1 is a block-diagram of a data transmission system including four LANs interconnected by a hub according to the principles of the invention.

Fig. 2 is a block-diagram of an ATM crossbar switch used within the hub illustrated in Fig. 1.

Fig. 3 is a time diagram representing the main signals exchanged between the ATM crossbar switch illustrated in Fig. 2 and the LAN adapters according to the invention.

Fig. 4 is a block-diagram of a LAN adapter within a data transmission system according to the invention.

Fig. 5 is a time diagram representing the main signals exchanged in a LAN adapter according to the invention.

Detailed description of the invention

The invention is implemented in an environment illustrated in Fig.1 wherein a plurality of Local Area Networks (LAN) 10, 12, 14, 16 are interconnected together by a hub 15 including an ATM crossbar switch 18 and the same plurality of LAN adapters.

Each LAN is connected to switch 18 by means of a LAN adapter 20 for LAN 10, 22 for LAN 12, 24 for LAN 14 and 26 for LAN 16.

In reference to Fig. 2, an ATM crossbar switch used in the invention includes a data switch module 30, a scheduler 32, a plurality of LAN adapter connectors for connecting a same plurality of LANs to the ATM crossbar switch, only two LAN adapter connectors 34 and 36 being represented, and a clock generator 38 for supplying the clock and the synchronization to data switch module 30, to scheduler 32 and to the LAN adapter connectors.

Data switch module 30 includes a switching data block 40 which is generally a passive switching matrix between data input signals from the LAN adapters to the switching matrix and data output signals from the switching matrix to the LAN adapters. It also includes a control logic 42 which decodes the configuration signals received from scheduler 32 to determine the data path connections and establishes the data path connection based on the synchronization signal received from clock generator 38.

The scheduler 32 comprises a control logic 44 receiving a request signal (REQ) from each LAN adapter and generating a grant signal (GNT) to each LAN adapter. A request signal is activated by a LAN adapter when it has a LAN frame to transmit to another LAN adapter through the ATM crossbar switch. Such a request signal is a serial encoded signal as explained below. When receiving the grant signal from the scheduler control logic 44, the LAN adapter transmits its frame .

The scheduler 32 also includes an algorithm unit 46 which determines the best data connection to establish at each time. Such a determination is based on the selection of the request amongst all requests received from the LAN adapters which meets some predetermined criteria such as a priority order,

the selection of unicast/multicast, the selection between reserved bandwidth data and non-reserved bandwidth data or any other criteria defined by the user.

The relationship between the signals at the interface between the ATM crossbar switch and LAN adapters are illustrated in Fig.3. First, the data clock pulses are used to exchange the LAN frames between the adapters through the switch card. 53 clock pulses determine the time slot to exchange 53 data bytes corresponding to the ATM cell size. It must be noted that there is no requirement on the clock rate.

The SYNCHRO signal is a one pulse clock during the first data byte of each time slot. The REQ signal is active during the first two data bytes of a time slot. Then, the algorithm processing occurs during the following 50 bytes of the slot. Finally, the GNT, when delivered for this REQ signal, is a one pulse clock during the 53rd data byte of the slot. Thus, in Fig. 3, a request X for tansmitting n slots is received at the beginning of the slot. Only one slot of data Z resulting from the preceding request is being transmitted during this slot, and therefore the grant signal for X is received at the end of the slot. Then, the corresponding data are transmitted during the following n slots. Note that during the last slot of this sequence of n slots, a new request Y is received and granted for the transmission of the corresponding data during the subsequent slots.

The REQ signal generated by each LAN adapter requesting to transmit a frame to the switch card is a serial encoded signal during the first two bytes of a time slot and includes 32 bits which are sampled by a signal the frequency of which is the data clock multiplied by 16. The first byte of the REQ signal includes the routing destination address on 16 bits, one bit per LAN adapter, a bit being set when the destination address corresponds to the associated LAN adapter. This encoding

scheme allows either a point-to-point connection, a multicast connection or a broadcast connection. The second byte of the REQ signal contains the connection time on 16 bits, that is the number of time slots required to transmit the entire frame.

It must be noted that, at each synchronization pulse, the control logic 44 of scheduler 32 (see Fig. 2) stores the 32 bits contained in the REQ signal from all LAN adapters. Then, algorithm unit 46 determines the best connection, sets the configuration data lines for the switch module 30 and activates the GNT signal to the selected LAN adapters. This new matrix switching state is latched up into switching matrix 40 on the falling edge of the GNT signal. This is done by control logic 42 of switching module 30.

In reference to Fig. 4, the hardware architecture of a LAN adapter is composed of a LAN logic 50 for processing the exchange of data with the LAN, a general bus 52 for transferring data bytes, a switch logic 54 for processing the exchange of data with the switch card, a system bus logic 56 for processing the transfer of data in the LAN adapter and an arbiter 58 for taking care of any bus contention for the requests which may come from LAN controller 64 or SCC 68 as well.

The LAN logic 50 comprises a LAN connector 60 allowing the connection of the LAN adapter to the LAN through a LAN attachment cable and carries the transmit data signal (TD) and the receive data signal (RD), an analog circuitry 62 for converting the TTL logic signals into analog and reciprocally and for providing specified network characteristics such as impedance, capacitance, crosstalk... LAN logic 50 also includes a LAN controller 64 which, when receiving a frame from LAN, performs the functions of :

- synchronizing its internal receive clock circuitry during the seven preamble bytes,
- detecting the LAN frame through the Start Frame Delimiter (SFD) byte,
- checking the data integrity of the frame by computing/comparing the four Frame Check Sequence (FCS) bytes,
- removing the protocol information such as preamble bytes, SFD byte and FCS bytes, and
- deserialising the remaining incoming bits to provide data bytes at the parallel interface with bus 52.

When transmitting data bytes from the parallel interface with bus 52 to the LAN, the LAN controller 64 performs the functions of :

- serialising the incoming parallel bytes,
- generating the protocol information bytes, and
- computing and sending the FCS bytes.

In the preferred embodiment the LAN controller 64 is a master device with an internal DMA controlling the transfer of bytes on the parallel interface with bus 52.

The switch logic 54 includes a switch connector 66, a serial communication controller 68 for transmitting serial to the switch card through connector 66 and receiving data from the switch card through connector 66, a control logic 70 generating the request signal and synchronizing the timing between the switch card and the LAN adapter, and a clock multiplier 72 for providing control logic 70 with the transmit clock generating the request signal at a frequency being 16 times the frequency of the data clock.

Connector 66 allows the connection of the LAN adapter to the switch card through a backplane and carries the request signal (REQ), the grant signal (GNT), the transmit data

signal (DATA OUT), the receive data signal (DATA IN), the data clock signal (DATA CLK) and the synchronization signal (SYNCHRO).

When transmitting data bytes from the parallel interface to the switch card, the serial communication controller 68 generates HDLC frames. It performs the functions of :

- generating an HDLC flag (one byte) to start a frame,
- serialising and sending the incoming parallel data bytes
- computing and sending the FCS (two bytes) after the data bytes, and
- generating an HDLC flag (one byte) to end the frame.

When receiving an HDLC frame from the switch card, the SCC (68) performs the hardware functions of :

- detecting the incoming frame through the flag,
- checking the data integrity of the frame by computing/comparing the Frame Check Sequence (FCS), and
- deserialising the incoming bits to provide data bytes at the parallel interface.

In the preferred embodiment, the SCC 68 is a master device with an internal DMA controlling the transfer of bytes on the parallel interface.

The system bus logic 56 includes a microcontroller 74 and a memory 76. Microcontroller 74 includes a processing unit, a ROS for storing the operational code, a RAM which operates like a cache memory and a programmable chip select for generating a memory chip select (CS1), a LAN controller chip select (CS2), a serial communication controller chip select (CS3) and a control logic chip select (CS4).

Memory 76 allows the LAN frame transfer between the LAN controller 64 and the serial communication controller. Such a memory is split in two independent areas, a LAN-to-switch

area organized in a first plurality of 2K bytes buffers and a switch-to-LAN area organized in a second plurality of 2K bytes buffers.

It must be noted that the general bus 52 is made of a data bus, an address bus and control signals such as read, write, chip selects, interrupts, bus requests and bus acknowledges, bearing in mind that the width of both data bus and address bus is not critical.

The operation of the system according to the invention is as follows. Following a machine power-on or a reset, microcontroller 74 initializes the three main functions of the LAN adapter illustrated in Fig. 4, that is the memory 76, the LAN controller 64 and the SCC 68. The initialization of LAN controller 64 consists in setting up the receive DMA of the controller with the base address of the LAN-to-switch buffer # 1 in memory 76. The initialization of SCC 68 consists in setting up the receive DMA of the SCC with the base address of the switch-to-LAN buffer # 1 in memory 76.

Assuming that a frame is received from the network on the receive line TD of connector 60, this frame is converted into TTL logic by analog circuitry 62 and transferred to LAN controller 64. While the incoming bits are stored in an internal receive FIFO, the receive DMA of LAN controller 64 requests the use of general bus 52 to arbiter 58 by activating the HOLD signal. When the general bus 52 is free, arbiter 58 activates the HLDA signal. From now on, the receive DMA of LAN controller 64 transfers the bytes of the frame from the FIFO of the LAN controller wherein they are stored into the LAN-to-switch buffer # 1 in memory 76. When the entire frame is stored in the memory, LAN controller 64 activates its interrupt signal INT1.

Wen receiving the interrupt signal INT1, microcontroller 74 stops its current task to execute a LAN interrupt routine by performing the actions of :

- reading the interrupt register of LAN controller 64 to determine the cause of the interruption,
- initializing the receive DMA of LAN controller 64 with the base address of the LAN-to-switch buffer # 2 in memory 76 (At this time a new frame coming from the network can be received),
- reading the frame byte count and the destination address, and
- jumping to a switch interface routine.

When running the switch interface routine, microcontroller 74 performs the actions of :

- determining the address of the destination LAN adapter using routing tables (it can be a unique address, a multicast address or a broadcast address),
- determining the Connection time (TC) by dividing the frame count by 53,
- storing both the destination address and the connection time in a parallel-to-series register located in control logic 70,
- initializing the transmit DMA of SCC 68 with the base address of the LAN-to-switch buffer #1 in memory 76 and the byte count, and
- starting the transmit DMA of SCC 68.

Then, the transmit DMA of SCC 68 requests the use of general bus 52 to arbiter 58 by activating its HOLD signal. When the general bus is free, arbiter 58 activates the HLDA signal. At this time, SCC 68 activates its Request-to-send line (RTS) to Control Logic 70 in order to transmit the frame to the switch card. When the Clear-to-send line (CTS) from Control Logic 70 is activated, the transmit DMA transfers the bytes from the LAN-to-switch buffer #1 of memory 76 into

the switch card. These bytes are sent in an HDLC format to guarantee the data integrity through the backplane. When the LAN-to-switch #1 is empty, SCC 68 activates its interrupt line INT 2. It must be noted that the HDLC format uses a flag when the end of the frame is reached even if the last slot is less then 53 bytes, and does not require the use of padding bits to complete a 53 bytes cells as in the ATM procedure.

Note that, it is the function of the Control Logic 70 to synchronize the timing of the different actions described above, such as outputting the destination address and the connection time on the request signal, getting the grant signal and setting up the CTS signal, with the timing of the switch card. This timing is illustrated in Fig.5.

When receiving the interrupt signal INT2 from SCC 68, microcontroller 74 stops its current task to execute a SCC interrupt routine by performing the action of reading the interrupt register of SCC 68 to determine the cause of the interruption and releasing the LAN-to-switch buffer #1 in memory 76.

Reciprocally, when SCC 68 detects the reception of a frame from the switch card, it requests the use of the general bus 52 to arbiter 58 by activating its HOLD line and stores the incoming bits in an internal receive FIFO. When the general bus is free, arbiter 58 activates a HLDA line to SCC 68. From now on, the receive DMA of SCC 68 transfers the bytes of the frame from the FIFO of SCC 68 into the switch-to-LAN buffer #1 in memory 76. When the entire frame is stored in memory 76, SCC 68 activates its interrupt line INT2.

When receiving the interrupt signal INT2, microcontroller 74 stops its current task to execute the SCC interrupt routine by performing the action of :

- reading the interrupt register of SCC 68 to determine the cause of the interruption,
- initializing the receive DMA of SCC 68 with the base address of the switch-to-LAN buffer #2 of memory 76 (At this time a new frame coming from the switch card can be received),
- initializing the transmit DMA of LAN controller 64 with the base address of the switch-to-LAN buffer #1 of memory 76 and the byte count, and
- starting the transmit DMA of LAN controller 64.

Then, the transmit DMA of SCC 68 requests the use of general bus 52 to arbiter 58 by activating its HOLD line. When the general bus is free, arbiter 58 activates the HLDA line to SCC 68. From now on, the transmit DMA of SCC 68 transfers the bytes of the frame from switch-to-LAN buffer #1 of memory 76 to the LAN. These bytes are transmitted serially through analog circuitry 62 onto the transmit line TD of connector 60. When the entire frame is sent out, LAN Controller 64 activates the interrupt line INT1 to microcontroller 74.

When receiving the interrupt signal INT1, microcontroller 74 stops its current task to execute the LAN interrupt routine by performing the actions of reading the interrupt register of LAN Controller 64 to determine the cause of the interruption and releasing the switch-to-LAN buffer #1.

CLAIMS

1. Data transmission system comprising a plurality of Local Area Networks LANs (10,12,14,16) interconnected by a hub (15) including a plurality of LAN adapters (20, 22, 24, 26) respectively connected to said LANs and an ATM crossbar switch (18) interconnecting all LAN adapters wherein at least one of said LANs wants to transmit a LAN data frame to another one of said LANs, said LAN data frame being converted into concatenated slots of the same size and being transmitted through said ATM crossbar switch ;

said system being characterized in that at least the LAN adapter connected to said LAN wanting to transmit the LAN data frame comprises a serial communication controller (68) including means for converting said LAN data frame received from said LAN into serial data having the form of concatenated slots of the ATM cell size in the HDLC format before transmitting said serial data to said ATM crossbar switch and means for converting serial data having the form of concatenated ATM cells received from said ATM crossbar switch into a LAN data frame before transmitting said LAN data frame to said LAN.

2. Data transmission system according to claim 1, wherein said LAN adapter further comprises a Control Logic (70) for generating a request signal (REQ) to said ATM crossbar switch (18) when said LAN wants to transmit LAN data frames to another one of said LANs

3. Data transmission system according to claim 2, wherein said request signal (REQ) is an encoded signal of 32 bits and wherein said LAN adapter further includes a clock multiplier for multiplying by 16 the data clock of the

system and providing said control logic with timing pulses used to transmit 32 bits of said request signal.

4. Data transmission system according to claim 3, wherein said request signal (REQ) includes a first data byte of 16 bits defining the destination address of said LAN data frames to be transmitted and a second data byte of 16 bits carrying the connection time defined by the number of slots to be transmitted.

5. Data transmission system according to claim 4 wherein said first data byte defining the destination address contains one bit for each LAN adapter, a bit being set when the destination address corresponds to the associated LAN adapter, allowing a point-to-point connection, a multicast connection or a broadcast connection.

6. Data transmission system according to any one of claims 1 to 5, wherein said serial communication controller (68) includes means for generating HDLC frames in response to said LAN data frames received from the LAN connected to said LAN adapter before transmitting said HDLC frames to said ATM crossbar switch (18).

7. Data transmission system according to claim 6, wherein said generating means in said serial communication controller (68) includes means for generating an HDLC flag to start a frame, means for serializing the incoming parallel data bytes, means for computing the FCS after the data bytes and means for generating an HDLC flag to end the frame.

8. Data transmission system according to any one of claims 1 to 7, wherein said serial communication controller (68) includes means for converting HDLC frames received from said ATM crossbar switch (18) into LAN data frames to be transmitted to said another LAN.

9. Data transmission system according to claim 8 wherein said converting means in said serial communication controller (68) includes means for detecting a starting HDLC frame in the incoming HDLC frame, means for checking the data integrity of the frame by computing the FCS, and means for deserializing the data bits of the frame to provide the data bytes of said LAN data frame.

10. Data transmission system according to any one of claims 1 to 9, wherein said LAN adapter further comprises a memory (76) split into two independent areas, a first LAN-to-switch area organized in a first plurality of buffers for storing said LAN data frames received from the LAN connected to said LAN adapter and to be transmitted to another LAN, and a second switch-to-LAN area organized in a second plurality of buffers for storing said LAN data frames received from another LAN and to be transmitted to the LAN connected to said LAN adapter.

11. Data transmission system according to any one of claims 1 to 10, wherein said LAN adapter further comprises a LAN controller (64) for converting said LAN data frame received in serial form from the LAN connected to said LAN adapter into parallel data bytes and reciprocally.

12. Data transmission system according to claim 11, wherein said LAN controller (64) comprises means for synchronizing its clock during the preamble bytes when receiving said LAN data frame, means for detecting the incoming frame through the delimiter byte of said frame, means for checking the data integrity of said frame by computing the FCS bytes, means for removing the protocol information of said frame, and means for deserializing the remaining incoming bits of said frames to provide parallel data bytes.

13. Data transmission system according to claim 11, wherein said LAN controller (64) further includes means for serializing the incoming data bytes received from said serial communication controller (68), means for generating the protocol information bytes to be included in said LAN data frame, and means for computing the FCS of said frame before transmitting said frame to the LAN connected to said LAN adapter.

14. Data transmission system according to claim 11, 12 or 13, further comprising an arbiter (58) for taking care of the contention between requests to send from said LAN controller (64) and requests to send from said serial communication controller (68).

15. Data transmission system according to any one of claims 1 to 14, wherein said ATM crossbar switch (18) comprises a scheduler (32) for enabling a request to transmit a LAN data frame from a LAN to another LAN to be granted or not.

16. Data transmission system according to claim 15, wherein said scheduler (32) includes an algorithm unit (46) for determining the best data connection to establish at each time based upon the selection of the request amongst all requests received from the LAN adapters which meets predetermined criteria.

SYSTEM FOR TRANSMITTING LAN DATA FRAMES THROUGH AN ATM CROSSBAR SWITCH

Abstract

Data transmission system comprising a plurality of Local Area Networks LANs (10,12,14,16) interconnected by a hub (15) including a plurality of LAN adapters (20, 22, 24, 26) respectively connected to the LANs and an ATM crossbar switch (18) interconnecting all LAN adapters wherein at least one LAN wants to transmit a LAN data frame to another LAN, the LAN data frame being converted into concatenated slots of the same size and being transmitted through the ATM crossbar switch. At least the LAN adapter connected to the LAN wanting to transmit a LAN data frame comprises a serial communication controller including means for converting the LAN data frame into serial data having the form of concatenated slots of the ATM cell size in the HDLC format before transmitting them to the ATM crossbar switch and means for converting serial data having the form of concatenated ATM cells received from the ATM crossbar switch into LAN data frames before transmitting it to the LAN.

FIG. 1

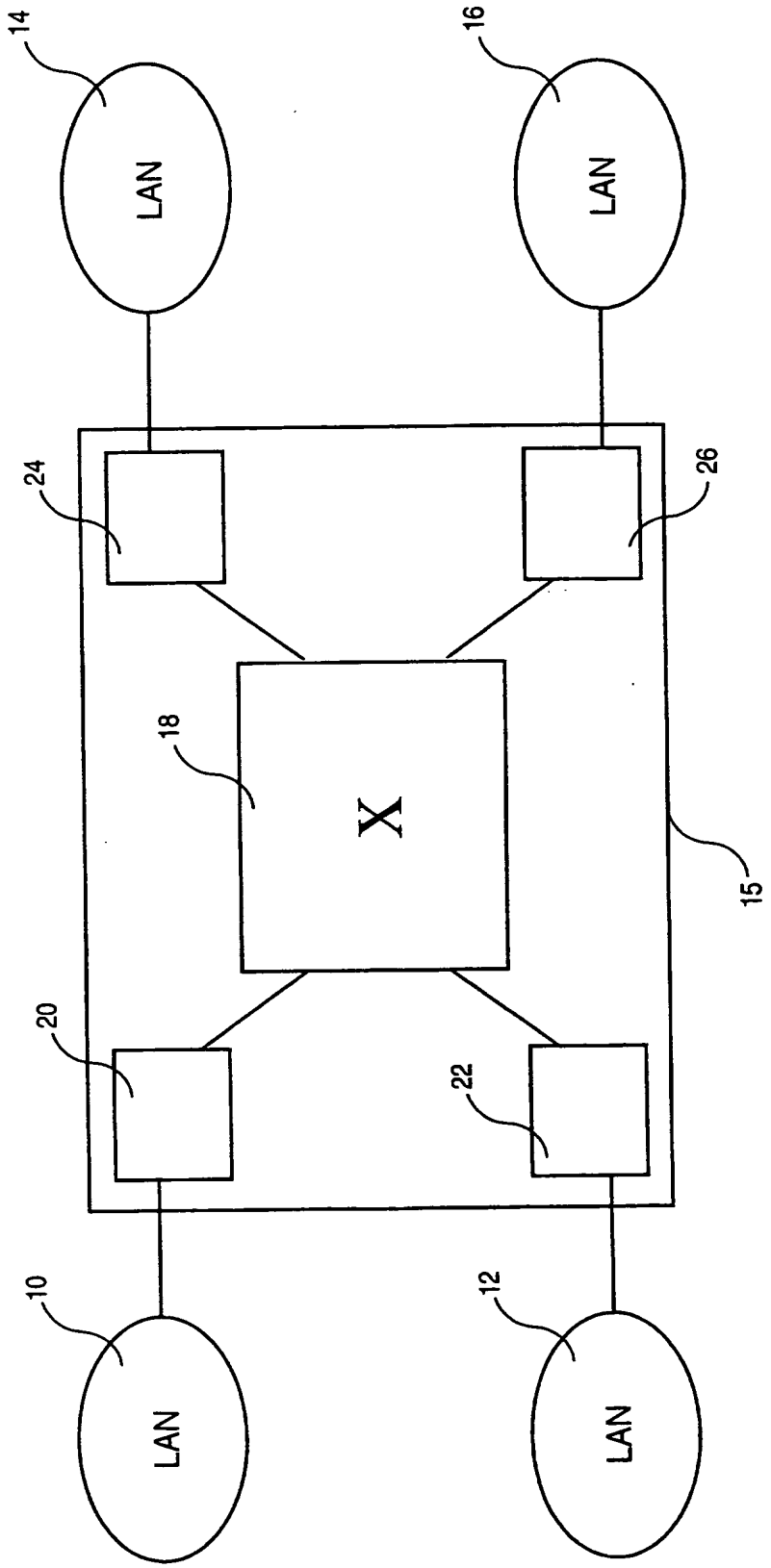

FIG. 1

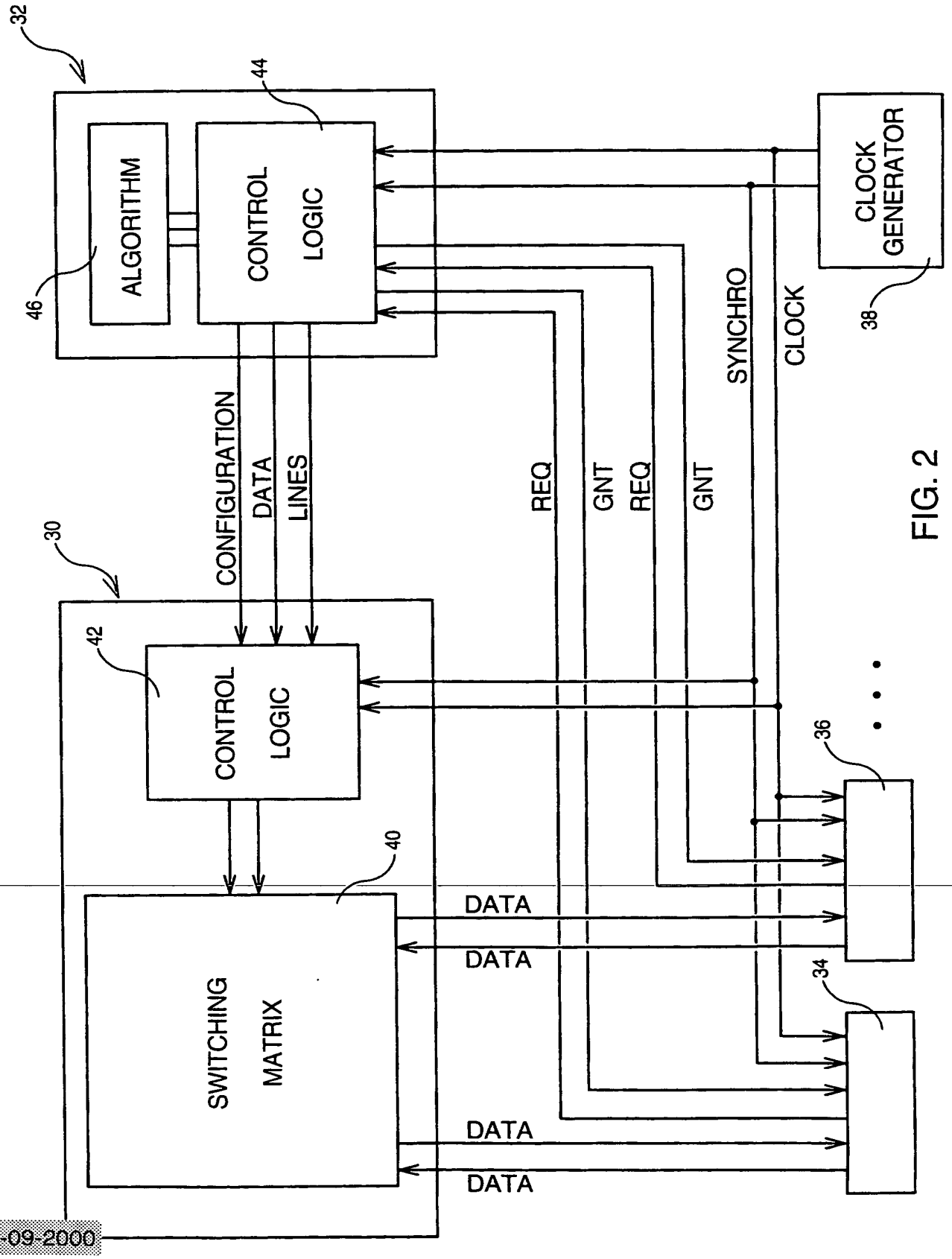
32
44
ALGORITHM
CONTROL LOGIC
46
CLOCK GENERATOR
38
SYNCHRO
CLOCK
CONFIGURATION DATA LINES
REQ
GNT REQ
GNT
30
42
CONTROL LOGIC
36
40
DATA
DATA
34
SWITCHING MATRIX
DATA
DATA

FIG. 2

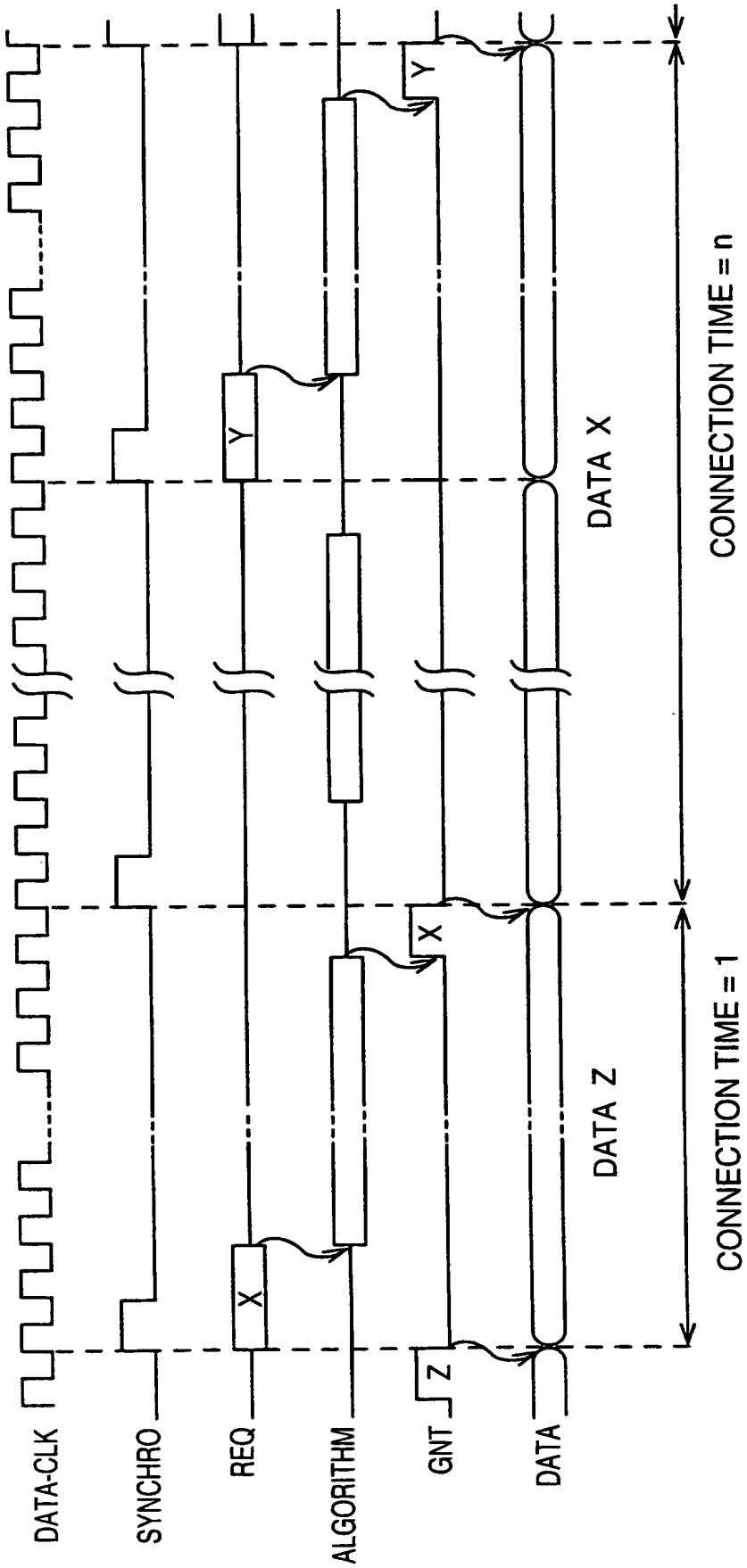
DATA-CLK
SYNCHRO
REQ
ALGORITHM
GNT
DATA
X
Y
Z
Y
X
DATA X
DATA Z
CONNECTION TIME = n
CONNECTION TIME = 1

FIG. 3

FR 9 00 026
Benayoun et al
4/5

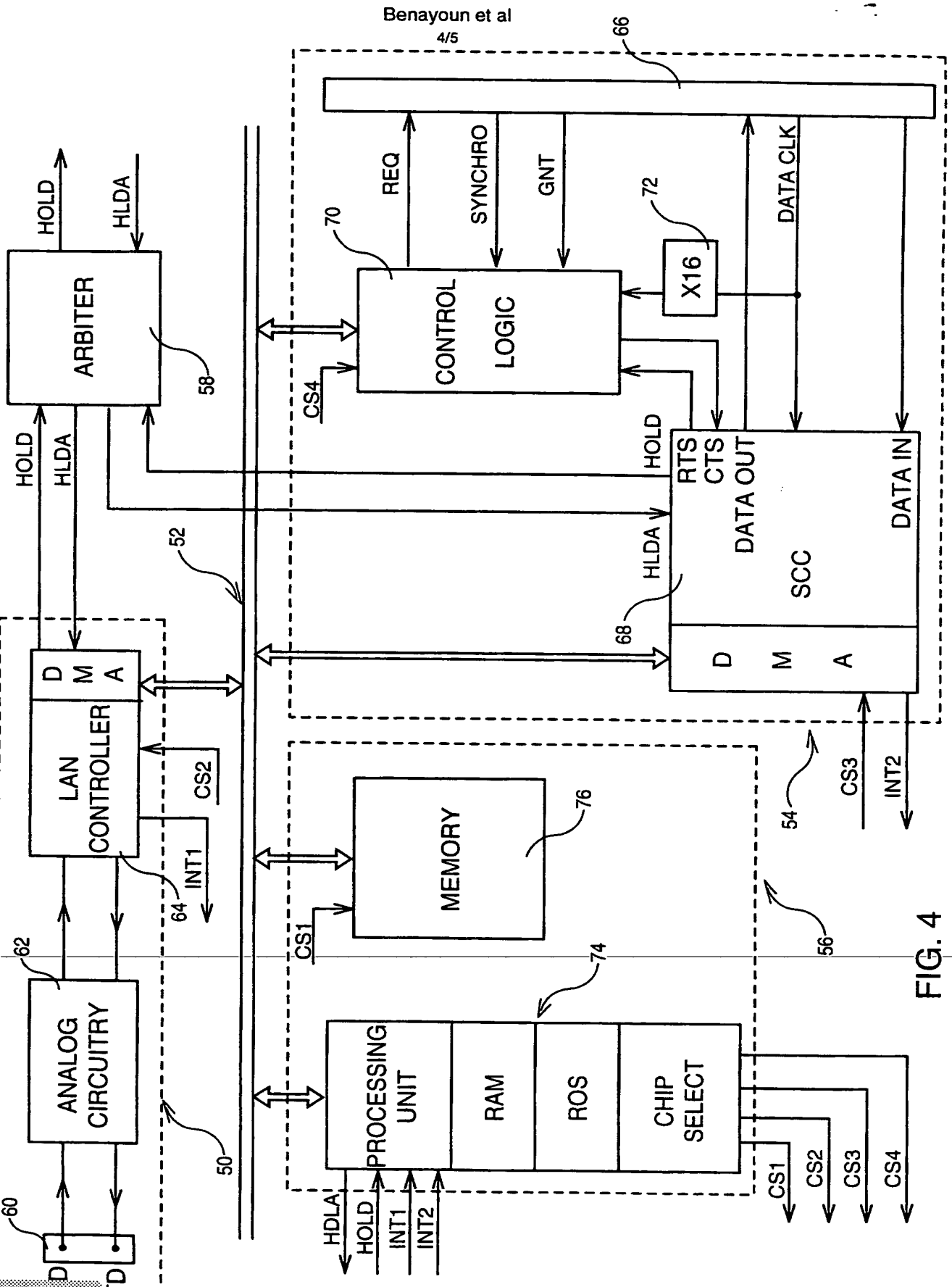

FIG. 4

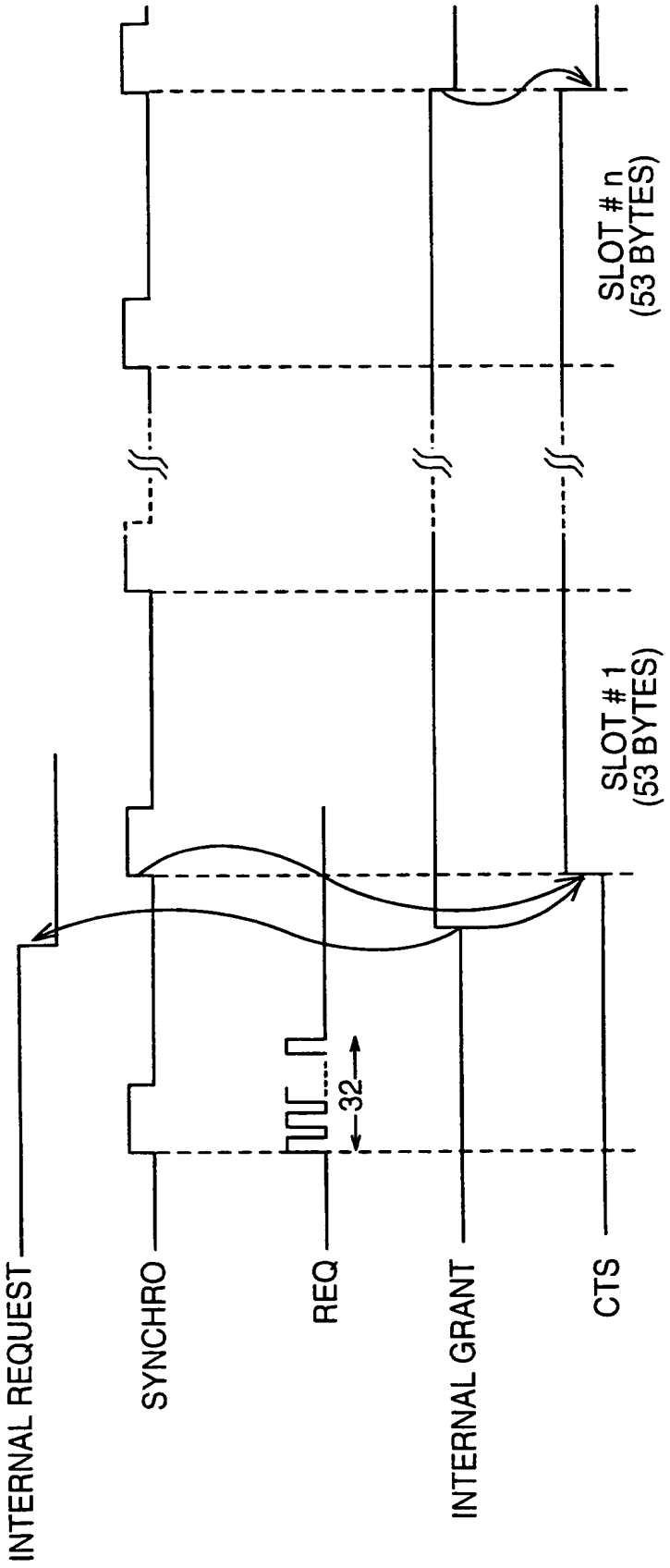
SLOT # n
(53 BYTES)
SLOT # 1
(53 BYTES)
32
INTERNAL REQUEST
SYNCHRO
REQ
INTERNAL GRANT
CTS

FIG. 5